《The logic route to strong AI》

Alibaba HKAI Lab 2020 presentation

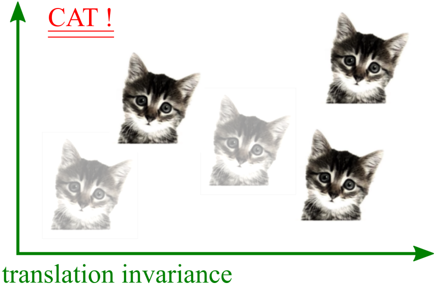
Heelal team

July 22, 2020

Table of contents

CNN 在机器视觉中的成功

- 在几何学上，视觉 具有 平移 不变性：



(1)

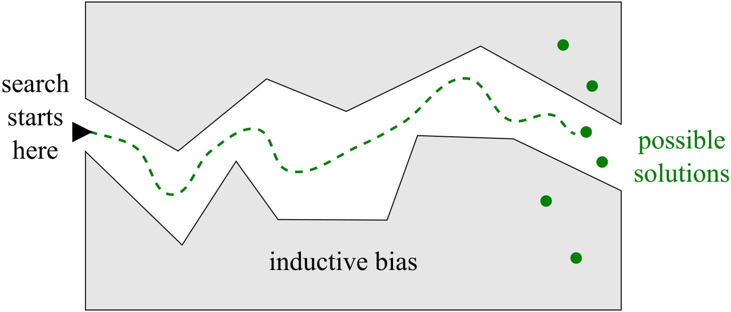
- Convolution 是一种具有平移不变性的运算：

$$(T_x \circ f) * g = T_x \circ (f * g) \tag{2}$$

- Yann LeCun 等人 利用 CNN 的 对称性 加快了学习速度，成功地解决了机器视觉 的问题

Symmetry and inductive bias

- 在数学上，对称性 经常能简化计算，所以数学家 特别喜欢 对称
- 在机器学习中，经常要引入 归纳偏好 (inductive bias)，缩小 搜寻空间：

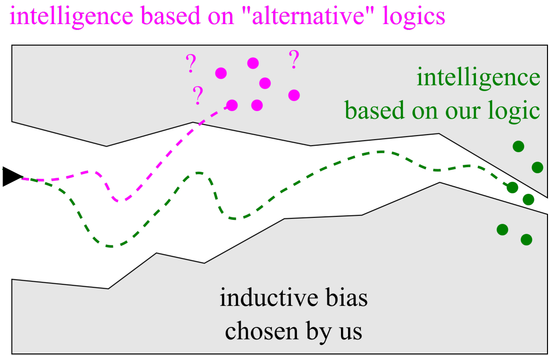


(3)

- 往往如果 归纳偏好 选对了，可以在短时间内找到答案，否则问题是不可解的 (intractable)

Richard Sutton 的观点

- Richard Sutton 认为，我们只需在 强化学习 的框架下 增加计算力，就可以找到 strong AI
- 下面描述的只是众多 形式逻辑 之中可能的一种：



(4)

- 这不只是一个「空想」的问题；事实上，世界各地的实验室 已经开始了对 AGI 不同形式的搜索！

对 逻辑主义 的质疑

- 很多人怀疑：人脑真的用 逻辑 思考吗？
- 其实我们每句表达的 语言，都是逻辑形式的 (logical form)
- 直觉认为，人脑 构造一些 models，再从 model 中「读出」一些结论
- 例如给定一个描述：「已婚妇人出轨，用刀刺死丈夫」

(5)

- 那么 妻子穿著什么衣服？衣服什么颜色？
- 这个 model 可以有哪些细节？答案是：任何细节都不可以有，除非是 逻辑上蕴含的
- 其实人脑可能比我们想像中更接近逻辑化的结构

Structure of logic

- 我的想法是：在深度学习中引入 逻辑 的对称性，解决 strong AI 问题
- 因为人的思维 具有 逻辑 的结构，这个 inductive bias 可以帮助我们快速找到 the solution to strong AI
- 逻辑结构很复杂，但最粗略的 symmetry 是 命题的 可交换律 (commutativity, or permutation invariance):

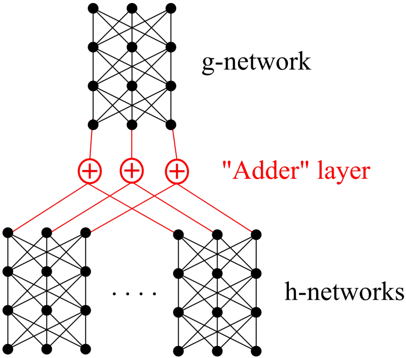
$$\begin{aligned} A \wedge B &\equiv B \wedge A \\ \text{下雨} \wedge \text{失恋} &\equiv \text{失恋} \wedge \text{下雨} \end{aligned} \tag{6}$$

- 它的重要性类似於 视觉中的 平移不变性
- 另一种讲法是：它将智能系统的 思维状态 (mental state) 分拆成 一粒粒独立的 命题 (propositions)

Symmetric neural networks

- Permutation invariance can be handled by symmetric neural networks
- 我浪费了两年时间试图解决这问题，却发现在 3 年前已经有两篇论文解决了 [PointNet 2017] [DeepSets 2017]，而且数学水平比我高很多！
- Any symmetric function can be represented by the following form (a special case of the Kolmogorov-Arnold representation of functions):

$$f(x, y, ...) = g(h(x) + h(y) + ...) \tag{7}$$



(8)

- Sym NN gives a powerful boost in efficiency $\propto n!$ where $n = \#\text{inputs}$
- The code for Sym NN is just a few lines of Tensorflow:

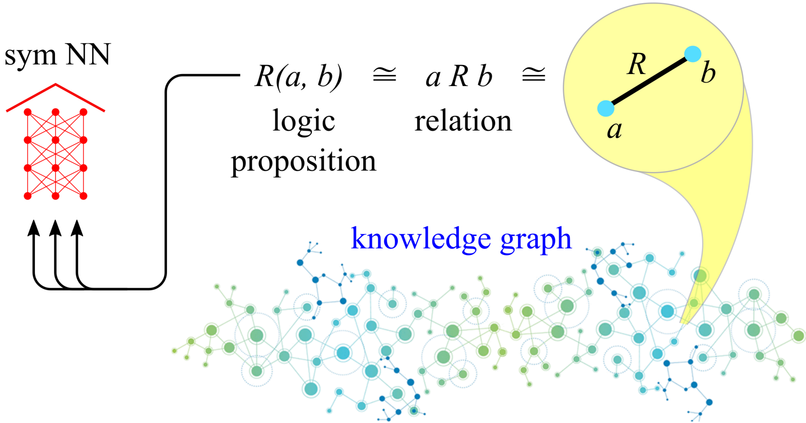
```
h = Dense(3, activation='tanh')
ys = []
for i in range(9):
    ys.append( h(xs[i]) )
y = Keras.stack(ys, axis=1)
Adder = Lambda(lambda x: Keras.sum(x, axis=1))
y = Adder(y)
g = Dense(3)
output = g(y)
```

(9)

- Very easy to adopt this to existing models such as BERT and reinforcement learning
- I have successfully tested it on the game of TicTacToe

应用：知识图谱 (knowledge graphs)

- 知识图谱 不能直接输入神经网络，它必需分拆成很多 edges，每个 edge 是一个 关系，也是一个 逻辑命题 ；也可以说 “graphs are isomorphic to logic”

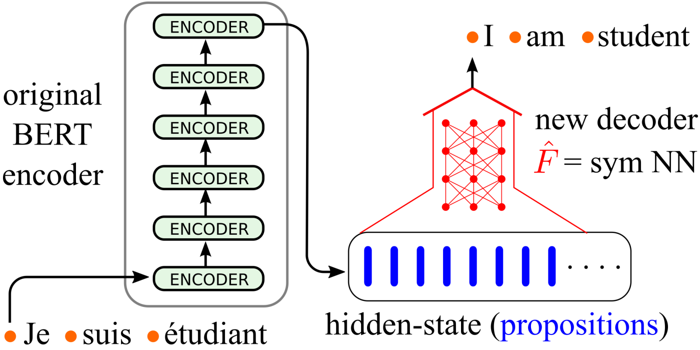


(10)

- 而这些 edges 似乎必需用 symmetric NN 处理，因为它们是 permutation invariant

应用：BERT 的逻辑化

- 类似地，可以将 BERT 的 隐状态 变成 “set of propositions” 的形式，方法是将 原来的 decoder 变成 sym NN：

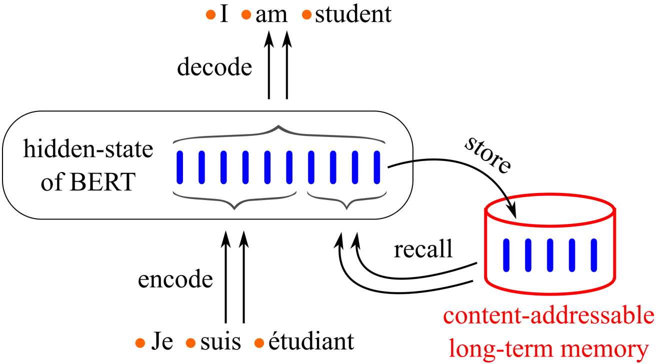


$$\text{(11)}$$

- 原来的 encoder 可以照旧使用，因为后半部改变了，error propagation 会令 representation 也改变
- 当然，这个想法有待实验证实 ☺️

应用：content-addressable long-term memory

- 将 BERT 的内部状态 逻辑化之后，可以储存在 长期记忆 中：

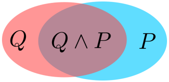


(12)

- 这种系统 已非常接近 strong AI
- Content-addressable memory 的想法来自 Alex Graves *et al* 的 Neural Turing Machine [2014]
- 以前 BERT 的隐状态 没有逻辑结构，我们不是很清楚它的内容是什么；这是有赖 逻辑化 才能做到的

再谈一次 逻辑结构

- 我发现 逻辑 和 AI 之间 有很漂亮的理论
- 现代逻辑理论 揭示出某种 几何 (geometry) 结构
- 大家 中／小学 时期 都熟悉 Venn diagrams，其实 命题逻辑 同构于 拓扑 (topology) 结构：



(13)

- 另方面，著名的 Curry-Howard isomorphism 揭示 逻辑证明 与 编程语言 之间的深刻关系：

$$\begin{aligned} \text{逻辑} \quad &\Leftrightarrow \quad \text{type theory} \\ \text{逻辑证明} \quad &\Leftrightarrow \quad \text{programs} \end{aligned} \tag{14}$$

- 程式 是一些 函数，而 神经网络 也是非线性的函数映射，所以 神经网络 也对应于 逻辑推理

- BERT 似乎是在进行著 句子之间的变换，而这些句子其实是 word embedding 的 concatenation 而已，例如：

$$\text{苏格拉底} \cdot \text{是} \cdot \text{人} \xrightarrow{BERT} \text{苏格拉底} \cdot \text{会} \cdot \text{死} \tag{15}$$

  这个做法表面上很「粗暴」，但其实这个映射和 Curry-Howard 对应的映射，在结构上似乎是一样的
- 集合元素 a 是 $\exists x.P(x)$ 的证明，例如 Socrates 是 $\exists x.\text{mortal}(x)$ 的证明
- 当 神经网络 调教 某些元素的 映射 (mapping) 时，它同时在学习某个逻辑的 formula; 换句话说，逻辑 是几何空间中的映射
- 逻辑的 谓词 (predicates) 是在 基底元素空间上的一个 纤维丛结构 (fibration)，记作 $\begin{smallmatrix}\mathbb{E}\\ \downarrow\pi\\ \mathbb{B}\end{smallmatrix}$: 这是从 几何／拓扑 “借” 来的概念，演变成 topos（拓扑斯）理论
- 这套漂亮的理论 不是无用的；长远来说，它可以指导 深度学习 和 逻辑 之间的 融合，例如诠释一些 vectors 的逻辑含义

References

[1] Zaheer et al. "Deep sets". In: *Advances in Neural Information Processing Systems* 30 (2017), pp. 3391–3401.

Illustration credits:

- Translation invariance, from Udacity Course 730, Deep Learning (L3 Convolutional Neural Networks ▷ Convolutional Networks)
- Étale space, from Topoi – The Categorical Analysis of Logic [Goldblatt 2006]